Subject: Re: Whats up in opening files: Unix vs. Windows
Posted by Nigel Wade on Tue, 13 Mar 2001 11:00:41 GMT
View Forum Message <> Reply to Message

> I use this code on a Unix System and it has been working fine for 4
years.

> file_header={file_header, nblocks: 0L, ntraces: 0L, np: 0L, ebytes: 0L,
> tbytes: 0L, bbytes: 0L, transf: 0, status: 0, spare1: 0L}
> data_header={data_header, scale: 0, status: 0, index: 0, spare3: 0,
ctcount:
> 0L, lpval: 0.0, rpval: 0.0, lvl: 0.0, rvl: 0.0}

> ;----Open the data file
> openr, unit, infile, /get_lun ; Open file
> point_lun, unit, 0
> readu, unit, file_header ; Read the file header (only occurs once)

> help, file_header, /st

[code snipped]

> ** Structure FILE_HEADER, 9 tags, length=32:
> NBLOCKS LONG 3276800
> NTRACES LONG 16777216

> Should be more like 164 and 220.
> Is there a differnce in 0L in Windows and Unix ?? Or structures, or
anything
> lese?

Well, if we believe what RSI tell us then the size of LONG on all platforms is 32bits. But the byte ordering may not be the same.

I presume the data files are written big-endian and you are reading them on Windows which runs on little-endian hardware. If that's the case you will certainly need to byte-swap. But that won't get you the numbers you expect.

The value for NBLOCKS you have above is 3276800, which in hex is 0x00320000. Byte swapping gives 0x00003200 which converts back to 12800 decimal.

Similarly, NTRACES byte-swapped is 1 so there's maybe more going wrong than just endian-ness problems.

--

Nigel Wade, System Administrator, Space Plasma Physics Group,
University of Leicester, Leicester, LE1 7RH, UK
E-mail : nmw@ion.le.ac.uk
Phone : +44 (0)116 2523568, Fax : +44 (0)116 2523555

Subject: Re: Whats up in opening files: Unix vs. Windows
Posted by Paul van Delst on Tue, 13 Mar 2001 14:36:09 GMT
View Forum Message <> Reply to Message

Sean Heukels wrote:
>
> I use this code on a Unix System and it has been working fine for 4 years.
>
<code snipped>
>
> The only thing is that the header is not read properly. It crashes on
> file_header.ebytes, which is set to 0 (0*2=> array size ?? no way)
> A piece of what it does read:
>
> ** Structure FILE_HEADER, 9 tags, length=32:
> NBLOCKS LONG 3276800
> NTRACES LONG 16777216
>
> Should be more like 164 and 220.

Looks like a byte swap problem given that SWAP_ENDIAN(16777216L) = 1. Posting just the code for this sort of problem isn't enough for diagnosing your problem. On what system was the file created and on what system are you trying to read it. You say "Unix" but have you upgrading your machine(s) or OS in the last 4 years? E.g. going from 32 to 64 bit OS may cause problems. Or a different platform (e.g. IBM->SGI). Regardless, binary files aren't too portable - *especially* if you're writing/reading using a structure variable which may or may not be padded to certain byte boundaries (*Very* platform dependent) - the endianness notwithstanding. Same problem using derived types in compiled f90/f95 code too.

Personally I only use "regular" binary files for testing algorithms on my development platform and even then I always check for endianness - just in case (some code sticks around like sh*t to a blanket). For distribution, I (nearly) always use netCDF.

paulv

--
Paul van Delst A little learning is a dangerous thing;
CIMSS @ NOAA/NCEP Drink deep, or taste not the Pierian spring;
Ph: (301)763-8000 x7274 There shallow draughts intoxicate the brain,
Fax:(301)763-8545 And drinking largely sobers us again.
paul.vandelst@noaa.gov Alexander Pope.